

Software-Defined Networking Based Network Virtualization for Mobile Operators

Omer Narmanlioglu¹ and Engin Zeydan²

¹*O. Narmanlioglu is with P.I. Works, Istanbul, Turkey 34912.*

E-mail: omer.narmanlioglu@piworks.net.

²*E. Zeydan is with Türk Telekom Labs, Istanbul, Turkey 34770.*

E-mail: engin.zeydan@turktelekom.com.tr.

Abstract

Software-Defined Networking (SDN) paradigm provides many features including hardware abstraction, programmable networking and centralized policy control. One of the main benefits used along with these features is core/backhaul network virtualization which ensures sharing of mobile core and backhaul networks among Mobile Operators (MOs). In this paper, we propose a virtualized SDN-based Evolved Packet System (EPS) cellular network architecture including design of network virtualization controller. After virtualization of core/backhaul network elements, eNodeBs associated with MOs become a part of resource allocation problem for Backhaul Transport Providers (BTPs). We investigate the performance of our proposed architecture where eNodeBs are assigned to the MOs using quality-of-service (QoS)-aware and QoS-unaware scheduling algorithms under the consideration of time-varying numbers and locations of user equipments (UEs) through Monte-Carlo simulations. The performances are compared with traditional EPS in Long Term Evolution (LTE) architecture and the results reveal that our proposed architecture outperforms the traditional cellular network architecture.

Keywords: Software-Defined Networking, Network Virtualization, Virtualization Controller, Schedulers, Long Term Evolution.

1. Introduction

Developing new innovative solutions within current network infrastructure with respect to today's requirements is becoming difficult every day due to high complexity of networks [1]. It should be noted that even though the mobile technology is advancing rapidly, data transmission has been performed through the same backhaul since second generation of mobile technologies which is still valid in current Long Term Evolution (LTE) systems. With the advancements in LTE-Advanced and small cell technologies, backhaul of Mobile Operators (MOs) is expected to be similar to data centers with mesh network topologies. Taking into account these facts, the currently deployed network architecture of Evolved Packet System (EPS) used in LTE presents several drawbacks. Constantly deploying the infrastructure network equipments at high capacity is both costly and inefficient for MOs. With respect to this, virtualizing the currently on demand network infrastructure owned by infrastructure owners, Backhaul Transport Providers (BTPs), is crucial. Therefore, managing the dynamic network traffics resulting from users of MOs and handling the possible complex Service Level Agreements (SLAs) between MOs and BTPs via dynamic slicing becomes more important. However, current infrastructure deployment solutions cannot enable such virtualization option for MOs due to lack of proper usage of recent technological advancements in network virtualization. Therefore, MOs and BTPs are looking for new innovative solutions in order to overcome the increasing demands of these network dynamics [2].

Recent developments such as Software-Defined Networking (SDN), which is initially implemented using OpenFlow protocols, provides powerful and simple approaches to manage complex networks by creating programmable, dynamic and flexible architecture, abstraction from hardware and centralized controller structure. In addition to SDN, network virtualization has become one of the major recent innovations that can also provide flexible and scalable logical infrastructure to every organization. In respect to this, network virtualization with SDN is an important paradigm that ensures the efficient usage of network resources. It can provide several features such as sharing of resources by breaking down the larger ones into multiple virtualized pieces, isolation of resources for better monitoring of data privacy and interference-free network access among users, aggregation for combining smaller resources into a single virtual resource, dynamism for fast deployment and reliable scalability in order to deal with the users' mobility, ease in resource management

for debugging, testing and rapid deployment purposes [3]. On the other hand, developing appropriate scheduling mechanisms for resource allocation plays a fundamental role and help to meet quality-of-service (QoS) requirements of applications used by MOs' users. Depending on the application types (voice-over-IP (VoIP), video conferencing, streaming media etc.), the requirements differ, however, they can be mapped to common parameters such as minimum guaranteed data rate, transmission delay, jitter and packet loss rate. Therefore, combining the advantages of SDN for network virtualization with appropriate scheduling algorithms under the consideration of those QoS parameters for dynamic resource allocations plays a key role in satisfying the demands of users associated with MOs as well as of BTPs.

1.1. Related Works

Before the introduction of SDN, traditional QoS providing approaches such as Integrated Services (IntServs) [4] and Differentiated Services (DiffServs) [5], had been standardized. However, there have been several drawbacks of these approaches. IntServ, a fine-grained traffic control architecture, is only applicable for small scale networks. DiffServ, on the other hand, is coarse-grained and applicable for larger networks, but it can only provide predefined/static 64 different classes of traffic to be differentiated since DiffServ routers forward the packets based on 8-bit DS field in the Internet Protocol (IP) header [5]. This makes it hard for DiffServ to fine tune the QoS of separate flows. For example, the limit of DS field can be reached when four tenants each with sixteen application traffic types exists in the system. On the other hand, SDN can enable fine-grained tuning (e.g. rules defined per flow) based on the specific application or user needs without restrictions. Therefore, approaches utilizing more scalable techniques such as SDN can provide better QoS guarantees for big networks that have large coverage areas as in the case for MOs.

New cellular network architectures based on SDN paradigm have also been extensively investigated in the literature [6, 7, 8, 9, 10, 11, 12]. In [6], SDN architecture with four extensions to controller platforms, switches and base stations is proposed for cellular networks. The proposed SDN architecture helps to simplify the design and management in order to address major limitations of today's cellular network architectures. A novel architecture, namely SoftCell, supporting fine-grained policies for cellular core network is proposed in [7] with the usage of packet classification on access switches that are next to the base stations and aggregation of traffic along multiple

dimensions. In [8], software-defined based mobile network architecture that increases the operator's innovation potential is presented and validated by testbed implementation. [9] provides techno-economic analyses of two network scenarios which are software-defined non-shared and virtualized shared networks as well as comparisons with traditional networks. In [10], the authors point out applications of network function virtualization (NFV) and SDN while minimizing the transport network load overhead against several parameters (i.e., delays, number and placement of data centers etc.) as the function placement problem and aim to model and provide a solution for LTE mobile core gateways. [11] examines several implementation scenarios of SDN in mobile cellular networks and SDN's contributions to inter-cell interference management, traffic control and network virtualization domains are explained. SoftRAN [12] abstracts all base stations in a local area as a virtual big base station that is managed by a centralized controller to perform load balancing, resource allocation, handover etc. while considering global view of the network. Moreover, although network sharing in the context of relationship between third parties (e.g. Mobile Virtual Network Operators (MVNOs), vertical players) and MOs has been widely discussed, most of the related works are in the context of economic advantages, business requirements and operational benefits that network sharing can introduce [13, 14, 15, 16]. Recently, The 5th Generation Partnership Project (5GPP) has proposed some vertical use cases when envisioned 5G architecture (which can be owned by different entities such as mobile, transport or cloud infrastructure providers) provides infrastructure slices over the same physical infrastructure [17]. None of these approaches, however, consider the opportunity of applying both virtualization of mobile core/backhaul and as a consequence dynamic assignment of available evolved Node-Bs (eNodeBs) to different MOs based on their traffic demands which come basically from their respective user equipments (UEs) using various scheduling algorithms. This can be especially achieved by designing a SDN-based shared EPS architecture for multiple MOs with a virtualization controller that is subject to instructions from BTP.

Many vendors such as Cisco, VMware, Big Switch, NEC etc. provide different approaches to network virtualization which are critical for infrastructure providers [18]. Additionally, different network virtualization technologies have been studied in the literature [19, 20, 21, 22]. In [19], the authors have classified different network hypervisors based on centralized and distributed architectures at the top level classification criterion and a second-

level classification is executed based on the hypervisor execution platform. Technologies including OpenVirteX [20] and FlowVisor [21] act as the transparent proxy between multiple controllers and forwarding elements that can create multiple slices of network resource based on different slicing dimensions such as bandwidth, topology, forwarding table or device central processing unit (CPU). FlowVisor is one of the enforcers of SDN based network virtualization leveraging the capabilities of OpenFlow in order to provide network isolation between different slices. OpenVirteX (OVX) is another network virtualization platform developed by Stanford University's ON.LAB. OVX provides a different perspective and approach to FlowVisor. It also provides alternatives to virtual addressing strategies in order to keep spacing and separation among tenants, to virtual network topology so that tenants can be able to define their respective architectures while ensuring resilience for underlay networks. For supporting additional failover capabilities in case of congestion and failures within the network, VeRTIGO, which is an extension of FlowVisor, has been proposed in [22]. Even though all those approaches have different capabilities and present different performance-complexity trade-offs, generally using any of those approaches as a virtualization controller is able to meet the virtualization requirements of our proposed SDN-based shared EPS architecture.

1.2. Our Contributions

In this paper, we develop an SDN-based virtualization controller architecture through a systematic modeling of virtual core and backhaul elements based on SDN paradigm. In our developed scenario, a network virtualization controller, which is owned by BTP, is directly connected to SDN controllers of MOs. After achieving the virtualization of core/backhaul network equipments, all eNodeBs associated with different MOs become a part of resource allocation problem for BTPs. This virtualization controller connected to backhaul and core network elements is used to adaptively perform eNodeB assignment to each MO under the consideration of time-varying numbers and locations of associated UEs and MOs' demands. Using the proposed architecture, problem of eNodeB assignment from an eNodeB pool to each MO is investigated. It should be noted that complex SLAs between MOs and BTP can yield the necessity of solving multi-objective optimization problem. In order to approach this complex problem, we simply focus on solving one or two optimization parameters such as fairness, data rate and satisfied-MO-ratio. This optimization is performed with the use of QoS-aware schedulers includ-

ing Max-Min Fairness (MMF) and Rate Guarantee (RG) and QoS-unaware schedulers including Round Robin (RR), Blind Equal Throughput (BET), Maximum Throughput (MT) and Proportional Fair (PF) that are executed in virtualization controller. Moreover, their performances are compared with existing traditional EPS architecture in LTE through Monte-Carlo simulations. The results reveal that our proposed architecture outperforms the traditional EPS in LTE architecture depending on the selected scheduling method by BTP with respect to considered optimization parameter(s). Additionally, we try to generate our proposed architecture with the use of MMF scheduler on *de-facto* SDN emulator, namely Mininet [23], to showcase the benefits of applying scheduling algorithms to provide good service quality for QoS-guaranteed MOs. Our contributions in this paper can be summarized as follows:

- An SDN-based shared EPS architecture is proposed in order to bring substantial benefits to MOs and BTPs which are responsible for setting up and maintaining Evolved Packet Core (EPC), backhaul and Evolved Universal Terrestrial Radio Access Network (E-UTRAN) of LTE cellular systems.
- The performance of the proposed SDN-based shared EPS architecture is investigated in terms of fairness, data rate and satisfied-MO-ratio with the use of several QoS-aware and QoS-unaware schedulers. The performance improvements with respect to traditional EPS used in LTE cellular network architecture are demonstrated under the consideration of macro cell channel models with time-varying demands of UEs associated with different MOs.

The rest of the paper is organized as follows. In Section II, we present our proposed virtualized EPS architecture after describing the traditional EPS in LTE. In Section III, the analytical expressions of scheduling algorithms for eNodeB assignment to different MOs are given and the performance of proposed architecture is evaluated in Section IV. Finally, we conclude the paper in Section V.

2. System Model and Architecture

2.1. Traditional EPS architecture in LTE networks

The 3rd Generation Partnership Project (3GPP) has proposed EPS architecture which is all-IP based mobile network topology with less hierar-

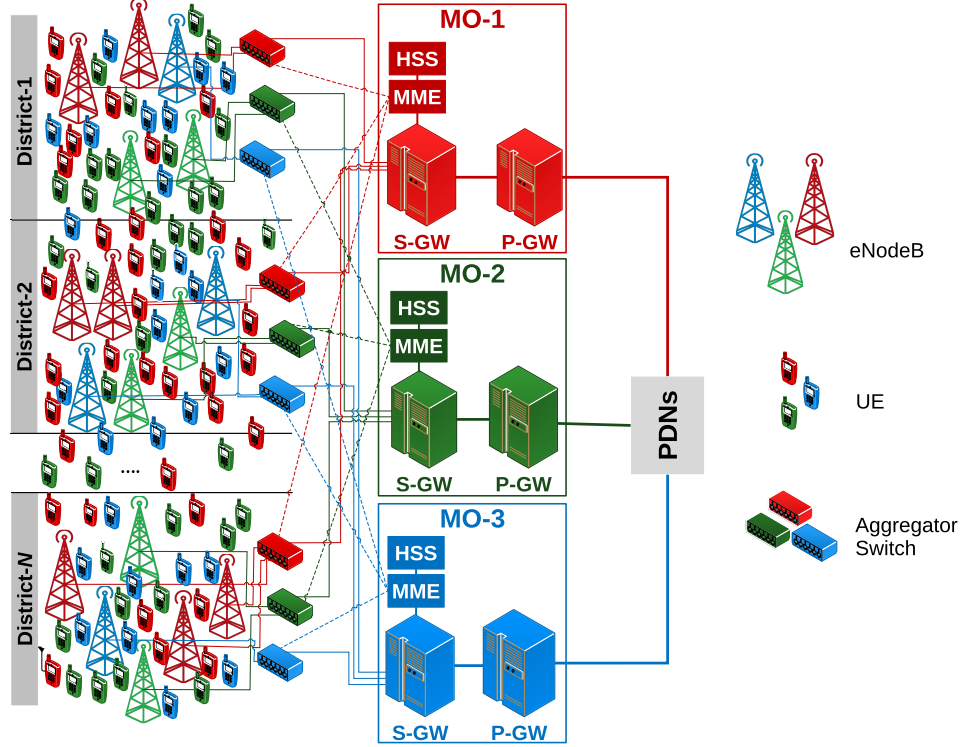


Figure 1: Traditional EPS architecture in LTE networks with three MOs.

chy and fewer nodes. EPC and E-UTRAN are the two main components of EPS. Base station and mobile terminal in LTE networks are denoted as eNodeB and UE, respectively. EPC includes: (i) Mobility Management Entity (MME) in which user mobility, tracking, paging, roaming and intra-LTE handover are performed, (ii) Serving Gateway (S-GW) that is responsible for routing and forwarding of packets among nodes and handover among LTE and other cellular networks, (iii) Packet Data Gateway (P-GW) that maintains the connection of LTE networks with the external IP-based networks, (iv) Home Subscriber Station (HSS) that provides user identification and addressing. On the other hand, E-UTRAN includes eNodeBs that handle the communication between UEs and core networks.

Fig. 1 shows an example of a traditional EPS architectural diagram of LTE networks for three MOs. These MOs are identified by different colors and they operate at the same geographical area where each MO has its own core, backhaul and radio access network elements or has rented some of

network equipments including mobile backhaul switches, routers from the main BTP (as in some countries). The network equipments of each MO (i.e., eNodeBs, aggregator switches, S-GWs and P-GWs) are also identified by the same colors and all of them are connected to some packet data networks (PDNs) such as Internet through their own infrastructure. It should be noted that in this architecture, none of the MOs are sharing any resource or equipment and each of them has deployed its own network equipments independent of each other.

The locations and numbers of eNodeBs as well as the capacity values of core/backhaul network elements associated with each MO are predetermined. This is done under the consideration of statistics of network characteristics such as average UE distributions, traffic loads, connection requests, etc. Thereby, they cannot be applicable for dynamic and adaptive changing network conditions which is one of the main drawbacks of this traditional EPS in LTE architecture. Moreover, due to the lack of dynamicity in the network, MOs may operate in over or under capacity conditions in some situations. As a consequence, this can introduce inefficient utilisation of network capacity, higher capital expenditure (CAPEX)/operating expenditure (OPEX) and connection problems during disastrous events.

2.2. Our Proposed Virtualized EPS Architecture

SDN allows the capability of virtualization based on different scenarios including topology, hardware, device CPU and bandwidth of individual links with priority settings within the network amongst MOs. In this section, we are introducing a new shared EPS architecture based on the SDN concept for mobile core/backhaul virtualization.

2.2.1. Virtualization Controller for Mobile Core and Backhaul Sharing

The network virtualization can readily apply to the provisioning of a SDN-based shared EPS network which is owned by BTP and utilized by MOs. The streams of different MOs are isolated from one another and each MO can control its own allocated slice of the network without any regard to the other MOs sharing the same network. The network slices allocated to individual MOs can be managed by the BTP via a virtualization controller (e.g., via a controller similar to FlowVisor [21]).

The SDN framework also allows the BTP to act as a broker in this setting to modify and adapt the slices in real time based on the agreements between the BTP and the MOs. The individual MOs can then control their own slices

via their dedicated control plane architectures (i.e., via their own MME, HSS and Policy and Charging Rules Function (PCRF)). Every time a new rule needs to be pushed by an MO's controller, the virtualization controller first checks the integrity and validity of the rule and then forwards the rule to the corresponding forwarders in the network. The SDN framework with the virtualization controller allows all nodes, including the network forwarding hardware and network gateways (S-GWs and P-GWs), PDNs and backhaul networks to be shared by MOs. It also provides granularity in what is shared in the network. In a shared network, the MOs may maintain their own eNodeBs, gateway elements, and PDNs or they may also share some of the gateway elements and PDNs. However, all MOs participating in the shared network have to maintain their own control planes (MME, PCRF and HSS) and those are used to control the network slices that are allocated by the virtualization controller which is managed by the BTP.

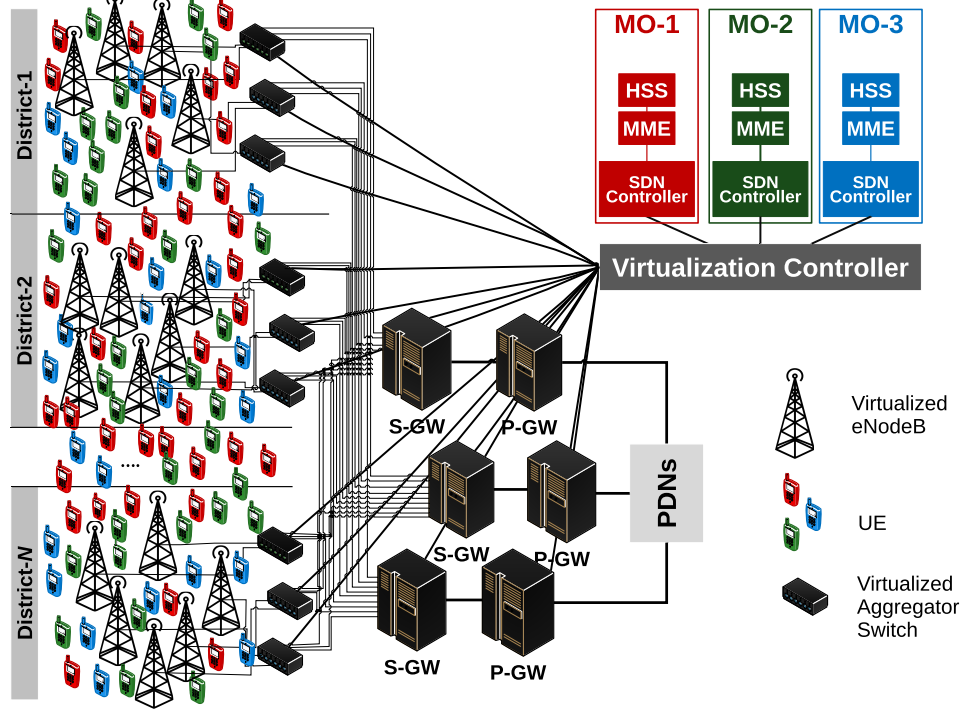


Figure 2: The shared SDN-based EPS architecture in LTE networks with three MOs.

In our scenario (see Fig. 2), a network virtualization controller that is directly connected to the SDN controllers of each MO is used to adaptively perform core/backhaul sharing between different MOs. Note that SDN controllers inside MOs control the demand requests of each user as well as establish bi-directional communications with virtualization controller. Since each eNodeB is connected to shared mobile backhaul and core gateways, such as S-GW via SDN-based aggregator switches, sharing of mobile backhaul/core network equipments ensures availability of pools of eNodeBs. The demands of multiple MOs are served under time and location based varying network conditions by pushing appropriate flow rules on both backhaul and core network elements via protocols such as OpenFlow. The districts of Figs. 1 and 2 demonstrate the situations where a pool of eNodeBs is available for multiple MOs in both traditional and proposed architecture. In the districts of Fig. 1, traditional eNodeBs assignment is depicted. When virtualization of core/backhaul network equipments is performed for each MO, all eNodeBs turn into a pool of radio access equipments owned by BTP (as in districts of Fig. 2). These radio access equipments can also be dynamically assigned to different MOs with respect to their number of active UEs, their varying demands and locations. This can provide many advantages such as efficient usage of network devices (e.g. availability of pool of eNodeBs), balancing traffic demand/usage behaviour via dynamic scaling of the network, automation of provisioning services and multi-tenancy for multiple MOs. In the shared mobile infrastructure, BTP, the infrastructure owner allocates the slices including eNodeBs and core/backhaul network equipments to MOs based on the assignment decision of virtualized eNodeBs. After that, all other radio access network (RAN) related eNodeB control functionalities including handover, roaming and radio resource management, carrier aggregation (CA) as well as interference management via Coordinated Multipoint (CoMP) transmission and reception, enhanced inter-cell interference cancellation (eICIC) etc. are managed by the virtualization controller of BTP, which is connected to the virtualized eNodeBs through the virtualized aggregator switches. The virtualization controller also communicates with associated MOs for those functionalities since all MOs still maintain their own control planes including MME, PCRF and HSS.

In this architecture, virtualization is performed at two levels. First, BTPs manage the network slices assigned to each MO using network virtualization controller. Second, sub-virtualization for all MO's applications is performed within a mobile operator's slice. In this SDN-based EPS architecture, traf-

fic of multiple MOs is converged to run on a common backbone network infrastructure while each stream of each MO is kept virtually separate. In E-UTRAN, all UEs are assumed to be under the coverage of multiple eNodeBs whose combination is abbreviated as District- i , for $i = \{1, 2, \dots, N\}$ (pool of eNodeBs), in Fig. 2 and they can be assigned to different MOs. Note that based on this scenario, sharing of mobile network backhaul equipments among multiple MOs and dynamic assignments of each eNodeB to different MOs through scheduling algorithms result in lower CAPEX and OPEX for both MOs and BTPs.

2.2.2. System Model and eNodeB Assignment Mechanism

A cellular network topology can be separated into several districts that do not overlap and do not interfere with each other. A single district is composed of a set ξ of eNodeBs, thereby, a set ξ of connections to an aggregator switch that is connected to S-GW and a set ζ of MOs which have different number of associated UEs and their time-varying demands, denoted by Ω , are considered. The performance parameters (i.e., data rate or capacity) give different values as a result of the connection and communication with MOs due to the variation in the total number of active UEs associated with these MOs and some physical factors such as path attenuation and fading. These factors lead to different signal quality at receiver side and different capacity levels on the links between eNodeBs and S-GW under the assumption that backhaul links are not bottleneck.

All possible links between eNodeBs and MOs can be modelled as a matrix, including all outcomes of set ζ (in row) and ξ (in column). When the matrix is denoted by $\mathbf{M} = (m_{i,k})_{|\zeta| \times |\xi|}$, the value of predefined performance parameter or metric with respect to the combination of the i^{th} MO and the k^{th} eNodeB is denoted by $m_{i,k}$. Each connection will be assigned to a specific MO and isolated from other MOs through a virtualization controller. Now, we define an assignment vector, Φ , whose length is the same with the set ξ and it includes the indices of MO associated with the connections. For instance, the vector of $\Phi = [2 \ 1 \ \dots \ 1]$ shows that the first connection (or eNodeB) is assigned to the second MO. Similarly, the second and last connections are assigned to the first MO. We are interested in finding the best assignment vector which provides the highest predefined performance parameters or metric(s) such as fairness, maximum data rate or any other parameter related to QoS requirements.

In this work, without loss of generality, we integrate the virtualization

controller, which is controlled by BTP explained in the previous section, into a single district for a general network topology. The virtualization controller aims at finding an assignment vector, Φ , using the various scheduling algorithms in our proposed architecture. The next section investigates those scheduling algorithms that are used by BTP throughout the proposed SDN-based EPS architecture.

3. Schedulers for Dynamic eNodeB Assignments in Shared EPS Architecture

Schedulers are the core component of resource management for optimization of the network performance. In this section, we briefly introduce their concepts and analytical expressions. Then, we provide a method to integrate those scheduling algorithms into the eNodeB assignment mechanism in our proposed architecture.

Basically, schedulers allocate the transmission resources to different users who have different QoS requirements and signal quality according to their allocation algorithms. The running algorithms in schedulers improve the system performance under the consideration of some performance metrics (i.e., total data rate, fairness among users) or QoS requirements. The algorithms have different input parameters, leading to different performances and system complexities. However, their allocation mechanisms are commonly based on giving the k^{th} resource to the i^{th} user, if its metric ($m_{i,k}$) satisfies the following equation,

$$m_{i,k} = \arg \max_j \{m_{j,k}\}. \quad (1)$$

In this paper, we classify the schedulers into three categories which are *channel-unaware*, *channel-aware* / *QoS-unaware* and *channel-aware* / *QoS-aware*. Table 1 shows two schedulers per category.

Table 1: Classification of considered schedulers.

	QoS-aware	QoS-unaware
Channel-aware	MMF, RG	MT, PF
Channel-unaware	—	BET, RR

Channel-unaware schedulers do not consider signal quality and QoS requirements. They use the simplest algorithms while allocating the resources

to users. RR and BET are *channel-unaware schedulers* which have fair allocation mechanisms. In RR, resource allocation is performed in a cyclic order of users and provides the fairness in terms of the number of allocated resources. Similarly, BET aims at satisfying the fairness in terms of data rate among users by using the past average data rate of each user. Its performance metric can be defined as inverse of $\lambda_i(t)$ where $\lambda_i(t)$ denotes the average data rate of the i^{th} user and it can be calculated by

$$\lambda_i(t) = \left(1 - \frac{1}{\tau}\right) \times \lambda_i(t - \Delta t) + \sum_k \frac{\delta_{i,k}(t - \Delta t) \times R_{i,k}(t - \Delta t)}{\tau}, \quad (2)$$

where $\tau > 1$ denotes time constant of smooth filter, Δt is the allocation interval (AI), which is the period of allocation, and $R_{i,k}(t)$ is the instantaneous achievable data rate of the i^{th} user with the use of k^{th} resource, and $\delta_{i,k}(t) \in \{0, 1\}$ takes the value of 1 if i^{th} user is allocated with k^{th} resource at t^{th} time, otherwise, it is 0.

In contrast, allocation mechanism of *channel-aware schedulers* depends on channel state information (CSI) that identifies the signal quality at receiver side and achievable data rates are estimated based on the periodically provided channel quality indicator (CQI) values. MT and PF schedulers fall into this category. The target of MT scheduler is to maximize the total data rate of the system by exploiting user diversity. Therefore, its metric can be written as

$$m_{i,k} = R_{i,k}(t). \quad (3)$$

However, it should be noted that MT is totally unfair since cell-edge users cannot be allocated with any resource due to bad-channel conditions. On the other hand, PF scheduler benefits from user diversity gain and considers proportional fairness among users. Therefore, PF partially satisfies both data rate and fairness. The allocation mechanism of PF can be thought of as a combination of BET and MT algorithms. Past average data rate as weighting factor is added into MT algorithm. The performance metric of PF algorithm for the i^{th} user can be written as

$$m_{i,k} = \frac{R_{i,k}(t)}{\lambda_i(t)}. \quad (4)$$

For generalized form of PF scheduler, it is defined as,

$$m_{i,k} = \frac{R_{i,k}(t)^\alpha}{\lambda_i(t)^\gamma}. \quad (5)$$

where α and γ are the weighting factors of data rate and fairness, respectively.

None of RR, BET, MT and PF algorithms take into account the QoS requirements. In general, the performance metric for QoS scheduler algorithms is written as [24]

$$m_{i,k} = R_{i,k}(t) \times U'(\lambda_i(t)), \quad (6)$$

where $U(\lambda_i(t))$ is defined as utility functions and $(.)'$ denotes the first order derivative operator. Utility function for QoS-aware users in RG scheduler [24], is designed as

$$U(\lambda_i(t)) = (\Omega_i) \times \left(\log(\lambda_i(t)) + 1 - e^{\left(-\beta_i \frac{\lambda_i(t) - \Omega_i}{\Omega_i}\right)} \right), \quad (7)$$

where β_i is a positive constant used to control the aggressiveness depending on the ratio of allocated resources and demands and Ω_i denotes the time-varying demand of the i^{th} user.

Another QoS-aware scheduler is MMF algorithm [25], in which resources are allocated to users orderly and with respect to their increasing demands (i.e., data rates) and unsatisfied users are equally allocated with the remaining resource.

To summarize all the scheduling algorithms discussed above, the pseudo code that finds the best assignment vector for each scheduling algorithm is given in Algorithm 1 where $\mathbf{R} = (R_{i,k})_{|\zeta| \times |\xi|}$ and $\mathbf{\Omega} = (\Omega_i)_{|\zeta| \times 1}$ denote the maximum achievable data rate matrix and demand vector respectively. In our case, allocated resources are eNodeBs and users are MOs. All scheduling mechanisms can be implemented in virtualization controller using this algorithm.

Algorithm 1: Scheduling algorithms running at virtualization controller that assigns eNodeBs to each MO.

Input: $\mathbf{R}, \mathbf{\Omega}$

Output: Φ

- 1 initialization: set Φ to zero;
 - 2 calculate \mathbf{M} ;
 - 3 **foreach** e **in** ξ **do**
 - 4 $\Phi(e) = \arg \max_{v \in \zeta} [m_{v,e}]$;
 - 5 **return** Φ ;
-

4. Performance Evaluations

In this section, we perform both simulation and emulation studies in order to investigate the performance of our proposed architecture with the use of several QoS-aware and QoS-unaware schedulers that are used for assignments of eNodeBs to MOs. In our proposed architecture, there exists a shared SDN-based EPS infrastructure in LTE networks with multiple MOs sharing resources from a pool of eNodeBs. Traditional cellular network architectures of EPS in LTE networks, where eNodeBs are statically assigned to each MO, are considered as benchmark.

We assume that there exist N non-overlapping districts (district-1, ... , district- N) connected to a single shared S-GW through aggregator switches. In our first simulation environment, UEs are uniformly and eNodeBs are deterministically distributed in the district-1 with the radius of 35 km, as shown in Fig. 3. Relative simulation parameters are given in Table 2. Demands of QoS-aware MOs which are MO-1 and MO-2 are uniformly distributed (unif) and MO-3 is considered as best-effort (BE) operator. In RG algorithm, β s for MO-1 and MO-2 are selected as 10 and 9.5, respectively (note that β is zero for MO-3). Additionally, time constant of smooth filter in BET, PF and RG schedulers are set to 50 and the generalized form of PF with α of 1 and γ of 0.8 is considered during simulations.

We consider two different static eNodeB assignments for MOs which are *demand-based* and *UE-based* assignments in traditional EPS architecture. In *demand-based* assignment (see Fig. 3 (a)), the numbers of eNodeBs associated with MO-1, MO-2 and MO-3 are set to 12, 18 and 1 in order to keep the proportionality with their demands. On the other hand, in *UE-based* assignment (see Fig. 3 (b)), the numbers of eNodeBs associated with MO-1, MO-2 and MO-3 are set to 9, 16 and 6 so that they remain proportional to the numbers of their associated UEs as in Table 2. In both *demand-based* and *UE-based* static assignments, eNodeBs are homogeneously distributed in the considered hexagonal district structure. In addition, our proposed architecture with virtualized eNodeBs is depicted in Fig. 3 (c).

Under the consideration of urban and suburban areas in macro cell structure and the carrier frequency of 2 GHz, the channel gain, denoted by H , between UEs and eNodeBs depending on path loss and shadowing effects can be calculated by

$$H = 128 + 37.6\log(d) + \psi \quad [\text{dB}], \quad (8)$$

where d is the distance between UE and eNodeB in km and ψ (in dB) is log-

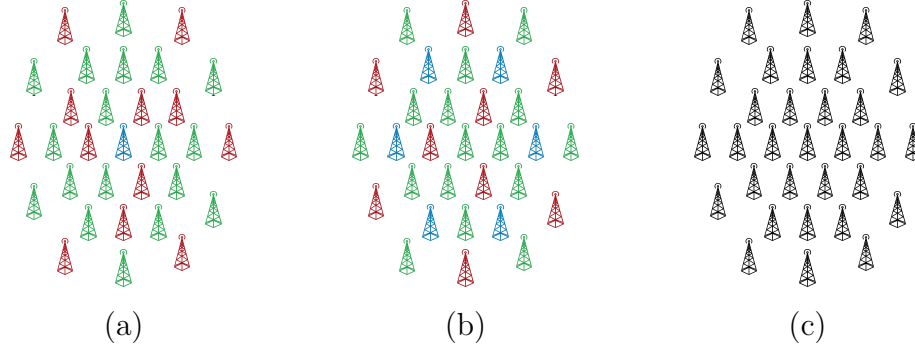


Figure 3: Locations of eNodeBs when (a) *demand-based* static assignment, (b) *UE-based* static assignment and (c) virtualized architecture are considered.

normal distributed (with zero mean and standard deviation of 8) shadowing effect. While calculating the maximum achievable data rate, we use Shannon Capacity (C) which can be written as

$$C = B \times \log_2 \left(1 + \frac{PH}{N_0 B} \right), \quad (9)$$

where B , P and N_0 are respectively bandwidth (Hz), transmit power (W) and noise power spectral density (PSD) (W/Hz) under the consideration of proper frequency spectrum sharing and advanced modulation techniques that ensure the interferences from neighbor eNodeBs to be insignificant. In order to obtain stable and confident simulation results, 1000 independent simulation runs are conducted. In each simulation, the locations of UEs are randomly selected. In addition to this, each independent simulation has 1000 time slots, in which shadowing effect and demands of users vary respectively in each and every fifty'th slots.

In Fig. 4, we show the average and the standard deviations of fairness values for traditional EPS in LTE architecture based eNodeB assignments with both *demand-based* and *UE-based* and our proposed architecture with the use of RR, BET, MT, PF, MMF and RG schedulers as detailed in Algorithm 1. Jain's fairness index is used as a fairness metric which can be defined as

$$\text{Fairness} = \frac{\left(\sum_{i=1}^M R_i \right)^2}{M \times \sum_{i=1}^M R_i^2}, \quad (10)$$

Table 2: Simulation parameters.

	MO-1	MO-2	MO-3
Number of UEs	300	500	200
Demand (Gbps)	unif(0, 8)	unif(0, 12)	—
Carrier frequency	2 GHz		
Bandwidth per user	5 MHz		
Transmit power	46 dBm		
Transmit power allocation	Uniform		
Noise PSD	−179 dBm/Hz		

where M is the total number of MOs.

Since BET scheduler uses past average data rate as an inversely weighting factor during allocation, MO with the highest data rate in current AI has less chance of obtaining resource in the next AI period. Therefore, BET satisfies fairness among MOs and presents the highest fairness index with the value of 0.99. Relatively, the other fair allocation algorithm, RR is the second with the highest value of 0.88 since it allocates the sources in a cyclic order. The other fairness based schedulers which are PF and MMF give 0.85 and 0.82 fairness indexes, respectively. When we turn to our benchmarks, *demand-based* assignment has 0.58 and *UE-based* assignment has 0.65 fairness values. The results show that our proposed architecture with RR, BET, PF and MMF outperforms both *demand-based* and *UE-based* assignments and it improves the fairness index. The reason for this improvement is due to the fact that the metrics of all four schedulers consider fairness issue and adopt the allocation mechanism with respect to time-varying UE locations while static assignments do not react against the time-varying factors.

Data rate performances of our proposed architecture and static assignments are depicted in Fig. 5. In contrast to fairness index, BET gives the lowest data rate with the value of 8.52 Gbps and this is followed by RR with 9.73 Gbps. MT scheduler algorithm, which gives the lowest fairness index of 0.37, achieves the highest data rate with of 14.68 Gbps. Additionally, RG scheduler gives 12.76 Gbps data rate. Under the consideration that the *demand-based* and *UE-based* assignments reach 12.06 Gbps and 11.21 Gbps, our proposed architecture shows improvement with the use of MT and RG schedulers when comparing both static assignments as the considered schedulers try to maximize the system throughput and satisfy the demands.

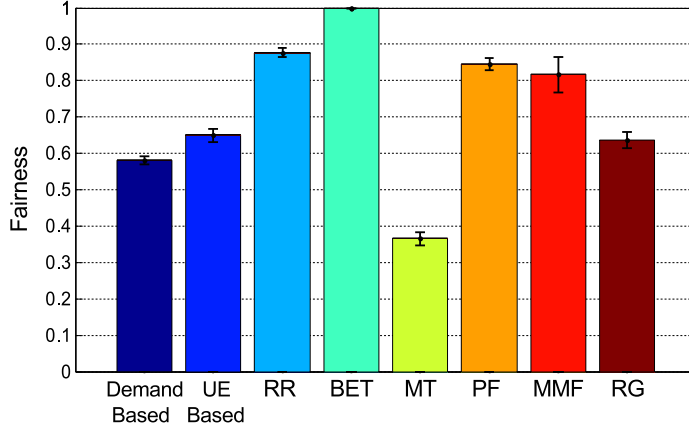


Figure 4: Fairness performances of traditional EPS in LTE (*demand-based* and *UE-based* eNodeB assignments) and proposed shared SDN-based EPS architecture with different scheduler algorithms.

In Fig. 6, we show the satisfied-MO-ratio performance. Since MT and PF schedulers consider CSI, MO-2 can be satisfied without considering QoS requirements, therefore, the ratios are 0.54 and 0.48, respectively. In MMF and RG with selected β values, satisfied-user ratios are 0.79 and 0.73 as *demand-based* and *UE-based* achieve the values of 0.61 and 0.51, respectively. These improvements through MMF and RG are due to the fact that while assigning eNodeBs to MOs, these allocation mechanisms consider QoS requirements of MOs.

Under the consideration of both fairness and satisfied-MO-ratio performances, our proposed architecture with the use of MMF scheduler outperforms the currently deployed architecture. Similarly, RG scheduler improves the performance when we consider both data rate and satisfied-MO-ratios. The results prove the benefits of programmable networking and centralized policy control (as in our proposed architecture) with respect to currently used traditional network architectures. The system performances in terms of fairness, data rate, satisfied-MO-ratio, both fairness and satisfied-MO-ratio and both data rate and satisfied-MO-ratios can be improved by the proposed architecture with a proper selection of scheduling algorithm such as MMF and RG respectively.

In Fig. 7, we present an example of eNodeB assignment results with the use of MMF scheduler in time varying channel and different demand

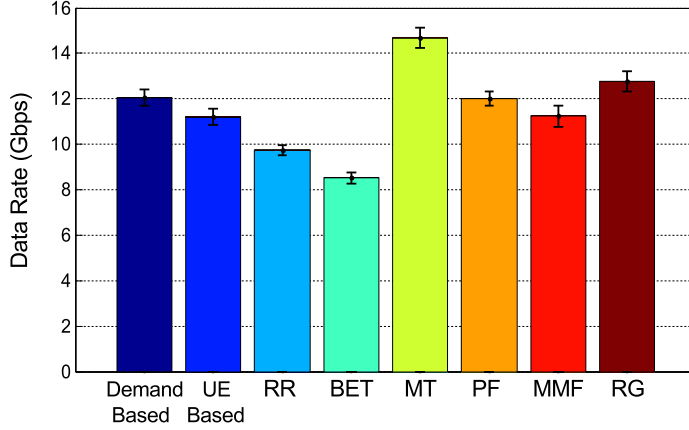


Figure 5: Data rate performances of traditional EPS in LTE (*demand-based* and *UE-based* eNodeB assignments) and proposed shared SDN-based EPS architecture with different scheduler algorithms.

conditions of each MO corresponding to the parameters defined above. In the time interval of 300 – 350, QoS-aware MOs which are MO–1 and MO–2 are satisfied and remaining eNodeBs are assigned to MO–3. Therefore, satisfied-MO-ratio reaches the highest value. On the other hand, because the demand of MO–2 is lower than that of MO–1 in the time interval 750 – 800 and available resource is not enough to satisfy both, only MO–2 achieves its demand. Consequently, MO–3 is not assigned with any eNodeBs by the BTP. Similarly, when the time interval of 100 – 150 is considered, since none of MO–1 and MO–2 are satisfied, MO–3 cannot serve its UEs.

In the second experimental results, we try to generate our proposed network architecture with the use of MMF scheduler on Mininet, *de-facto* SDN emulator. The generated network topology is depicted in Fig. 8 where three hosts, three OpenFlow switches, video servers and a virtualization controller exist. The hosts are assumed to be MOs as in our proposed architecture and it is investigated whether their demands are satisfied or not after they send a request to video servers. The video data is delivered to hosts through P-GW, S-GW and aggregator switch that are virtualized through FlowVisor as in SDN-based LTE architecture. The multiple links between aggregator switch and S-GW are likened to eNodeBs since eNodeBs direct the data traffic from UE to S-GW and vice versa over GPRS Tunneling Protocol (GTP) tunnel in LTE networks. After virtualization of both aggregator switch and S-GW,

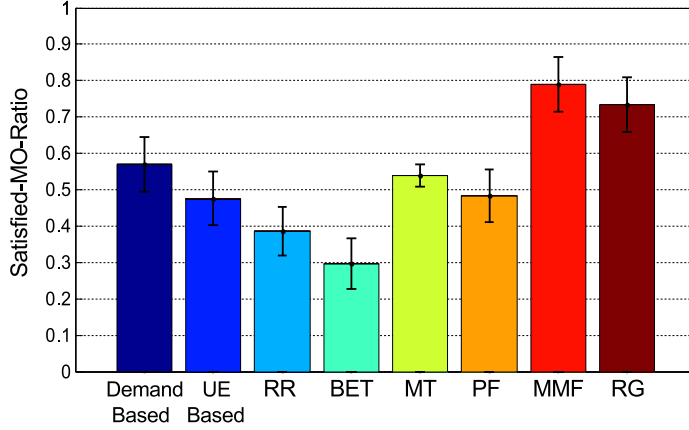


Figure 6: Satisfied-MO-ratio performances of traditional EPS in LTE (*demand-based* and *UE-based* eNodeB assignments) and proposed shared SDN-based EPS architecture with different scheduler algorithms.

link assignment is performed through FlowVisor virtualization controller, similar to eNodeB assignment. Wireshark[®] is used for further analysis such as traffic monitoring and collection of the amount of transferred data. The requirement of video transfer is identified as approximately 3 Mbps. QoS-aware users (MO-1 and MO-2) are assigned with links responding to their demands with the use of MMF algorithm and the remaining links in between aggregator switch and S-GW (whose total capacity is less than 3 Mbps) are assigned to BE user (MO-3). Under the consideration of assigned links, video qualities are presented in Fig. 9. Since QoS-aware users are satisfied, their video quality is good enough, however, the video of BE user is fully defective due to insufficient assignment.

5. Conclusion and Future Work

A detailed analysis for designing a virtualization controller which is controlled by BTP in a shared SDN-based EPS architecture that can benefit both MOs and BTPs has been provided. The eNodeBs become a part of resource allocation problem for BTP as a consequence of core and backhaul network virtualization and their assignment to different MOs are performed by BTP based on the demands of the MOs and the adopted scheduling algorithms. The results reveal that depending on the selection of scheduling

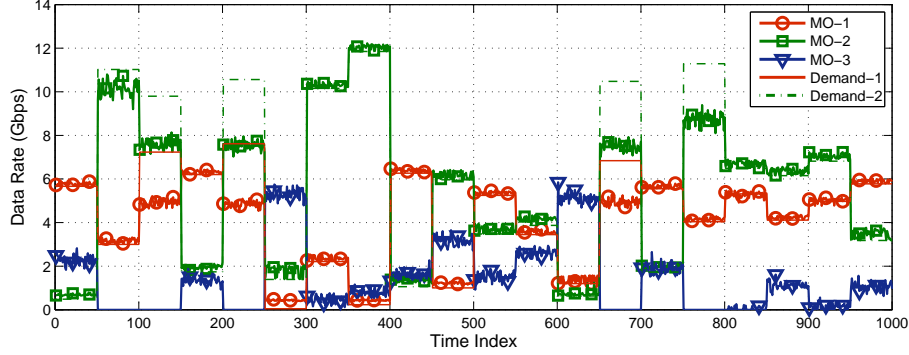


Figure 7: Evolution of obtained data rates three MOs as well as demands of two MOs with the use of MMF scheduler by BTP.

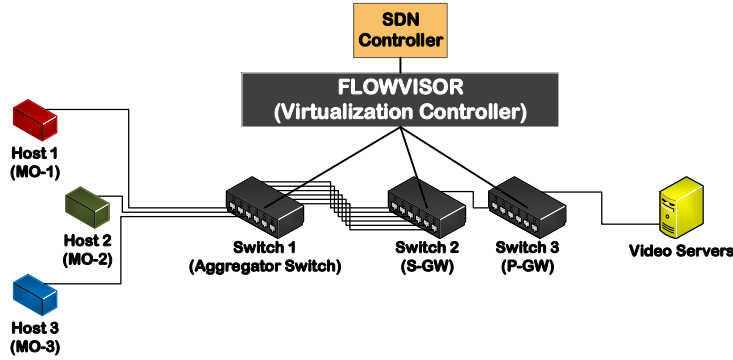


Figure 8: Considered topology on Mininet.

algorithm, our proposed SDN-based EPS architecture outperforms the traditional EPS with the use of both QoS-aware (MMF and RG) and non-QoS-aware (RR, BET, MT and PF) schedulers in terms of fairness, data rate and satisfied-MO-ratio, and this shows the benefits of programmable networking and centralized policy control. In addition, in order to showcase a real emulation environment of proposed architecture, the performance of MMF scheduling algorithm is demonstrated using the Mininet emulation platform in terms of video quality for different MOs.

Possible future extensions of this work include considering SLAs between different MOs when BTPs are running them over the same infrastructure. In those scenarios, due to existence of more than two conflicting objective func-

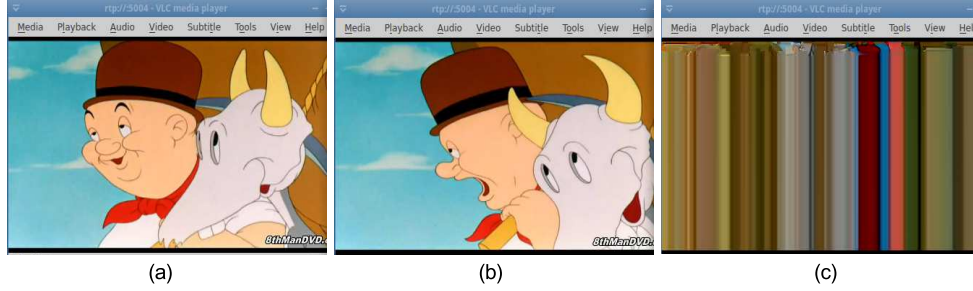


Figure 9: Validations on Mininet for video quality on (a) MO-1, (b) MO-2 and (c) MO-3.

tions, multi-objective optimization techniques leading to Pareto optimality, instead of the simpler QoS-based algorithms, are of interest.

Acknowledgment

This work has been performed in the framework of the Celtic-Plus project C2012/2 – 5 SIGMONA and TÜBİTAK TEYDEB (project no 9130038) project. The organizations on the source list would like to acknowledge the contributions of their colleagues in the project.

References

References

- [1] H. Kim, N. Feamster, Improving network management with software defined networking, *Communications Magazine*, IEEE 51 (2) (2013) 114–119.
- [2] S. Sezer, S. Scott-Hayward, P. Chouhan, B. Fraser, D. Lake, J. Finnegan, N. Viljoen, M. Miller, N. Rao, Are we ready for SDN? Implementation challenges for software-defined networks, *Communications Magazine*, IEEE 51 (7) (2013) 36–43.
- [3] R. Jain, S. Paul, Network virtualization and software defined networking for cloud computing: a survey, *Communications Magazine*, IEEE 51 (11) (2013) 24–31.

- [4] C. D. Braden, R., S. Shenker, Integrated Services in the Internet Architecture: An Overview, RFC 1633 (June 1994).
- [5] F. Baker, D. Black, S. Blake, K. Nichols, Definition of the Differentiated Services Field (DS field) in the IPv4 and IPv6 headers, RFC 2474 (Dec. 1998).
- [6] L. E. Li, Z. M. Mao, J. Rexford, Toward software-defined cellular networks, in: Software Defined Networking (EWSDN), 2012 European Workshop on, IEEE, 2012, pp. 7–12.
- [7] X. Jin, L. E. Li, L. Vanbever, J. Rexford, Softcell: Scalable and flexible cellular core network architecture, in: Proceedings of the ninth ACM conference on Emerging networking experiments and technologies, ACM, 2013, pp. 163–174.
- [8] K. Pentikousis, Y. Wang, W. Hu, Mobileflow: Toward software-defined mobile networks, Communications Magazine, IEEE 51 (7) (2013) 44–53.
- [9] B. Naudts, M. Kind, F.-J. Westphal, S. Verbrugge, D. Colle, M. Pickavet, Techno-economic analysis of software defined networking as architecture for the virtualization of a mobile network, in: Software Defined Networking (EWSDN), 2012 European Workshop on, IEEE, 2012, pp. 67–72.
- [10] A. Basta, W. Kellerer, M. Hoffmann, H. J. Morper, K. Hoffmann, Applying NFV and SDN to LTE mobile core gateways, the functions placement problem, in: Proceedings of the 4th workshop on All things cellular: operations, applications, & challenges, ACM, 2014, pp. 33–38.
- [11] S. Tomovic, M. Pejanovic-Djurisic, I. Radusinovic, SDN based mobile networks: Concepts and benefits, Wireless Personal Communications 78 (3) (2014) 1629–1644.
- [12] A. Gudipati, D. Perry, L. E. Li, S. Katti, SoftRAN: Software defined radio access network, in: Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking, ACM, 2013, pp. 25–30.

- [13] A. Khan, W. Kellerer, K. Kozu, M. Yabusaki, Network sharing in the next mobile network: TCO reduction, management flexibility, and operational independence, *IEEE Communications Magazine* 49 (10) (2011) 134–142.
- [14] T. Frisanco, P. Tafertshofer, P. Lurin, R. Ang, Infrastructure sharing and shared operations for mobile network operators from a deployment and operations view, in: *NOMS 2008-2008 IEEE Network Operations and Management Symposium*, IEEE, 2008, pp. 129–136.
- [15] C. Liang, F. R. Yu, Wireless virtualization for next generation mobile cellular networks, *IEEE Wireless Communications* 22 (1) (2015) 61–69.
- [16] K. Samdanis, X. Costa-Perez, V. Sciancalepore, From network sharing to multi-tenancy: The 5G network slice broker, *IEEE Communications Magazine* 54 (7) (2016) 32–39.
- [17] 5GPP, 5G empowering vertical industries, https://5g-ppp.eu/wp-content/uploads/2016/02/BROCHURE_5PPP_BAT2_PL.pdf, [Online; accessed -August-2016] (2016).
- [18] SDNCentral, LLC, 2015 Special Report: Network Virtualization in the Data Center (2015).
- [19] A. Blenk, A. Basta, M. Reisslein, W. Kellerer, Survey on Network Virtualization Hypervisors for Software Defined Networking, *Communications Surveys Tutorials*, IEEE 18 (1) (2016) 655–685.
- [20] A. Al-Shabibi, M. De Leenheer, M. Gerola, A. Koshibe, G. Parulkar, E. Salvadori, B. Snow, OpenVirteX: Make your virtual SDNs programmable, in: *Proceedings of the third workshop on Hot topics in software defined networking*, ACM, 2014, pp. 25–30.
- [21] R. Sherwood, G. Gibb, K.-K. Yap, G. Appenzeller, M. Casado, N. McKeown, G. Parulkar, Flowvisor: A network virtualization layer, OpenFlow Switch Consortium, Tech. Rep.
- [22] R. Doriguzzi Corin, M. Gerola, R. Riggio, F. De Pellegrini, E. Salvadori, VeRTIGO: Network Virtualization and Beyond, in: *Software Defined Networking (EWSDN)*, 2012 European Workshop on, 2012, pp. 24–29.

- [23] B. Lantz, B. Heller, N. McKeown, A network in a laptop: rapid prototyping for software-defined networks, in: Proceedings of the 9th ACM SIGCOMM Workshop on Hot Topics in Networks, ACM, 2010, p. 19.
- [24] K. D. Singh, D. Ros, Normalized rate guarantee scheduler for high speed downlink packet access, in: Global Telecommunications Conference, 2007. GLOBECOM'07. IEEE, IEEE, 2007, pp. 576–580.
- [25] D. Nace, M. Pióro, Max-min fairness and its applications to routing and load-balancing in communication networks: a tutorial, Communications Surveys & Tutorials, IEEE 10 (4) (2008) 5–17.

Omer Narmanlioglu received B.Sc. and M.Sc. degrees from the Department of Electrical and Electronics Engineering at Bilkent University, Ankara, Turkey, in 2014 and Ozyegin University, Istanbul, Turkey, in 2016. He is currently working with P.I. Works and pursuing PhD degree at Ozyegin University. His research interests are physical layer aspects of communication systems and software-defined networking paradigm for cellular networks.

Engin Zeydan received his PhD degree in February 2011 from the Department of Electrical and Computer Engineering at Stevens Institute of Technology, USA. He has worked as an R&D engineer for Avea, a mobile operator in Turkey, between 2011 and 2016. He is currently with Türk Telekom Labs. His research interests are in the area of telecommunications and big data networking.